

5/PRTK

10/509093

PCT/SE02/00618
DT04 Rec'd PCT/PTO 28 SEP 2004Title

Method and system for reserving resources within
an IP-network

Field of the invention

5 The present invention relates to an Internet Protocol (IP) network.

In particular, it relates to a method, system, and resource manager for reserving
resources within said IP network

10 Background of the invention

The primary goal when the Internet Protocols were designed was to provide an
effective technique for interconnecting existing networks. Other important goals
were survivability in case of failure and generality in supporting various services
and applications. To reach these goals, the IP protocol was designed to provide a
15 connectionless datagram network that does not require signalling and per-flow
forwarding state in network elements. It has turned out that the architecture scales
to large networks and supports applications making many end-to-end connections
(e.g. the World Wide Web).

20 One design trade-off made to enable interconnection was to support only best-effort
service at the network level and rely on endpoint functionality to obtain various
levels of service. Best-effort service provides adequate support for traditional data
applications that can tolerate delay, loss and varying throughput along the path.
However, in networks carrying high loads of traffic, this type of service is often
25 inadequate for meeting the demands of applications that are more sensitive to
packet loss and delay (e.g. telephony, video on demand, multimedia conferencing,
etc.).

Traditionally, demanding real-time applications have been built on networks that
30 are vertically optimised for the particular application. This design principle results
in networks that are efficient for their purpose, but do not easily support new
applications and are in many cases incapable of efficiently multiplexing applications
with varying resource demands. It has turned out that the cost of running several
different networks in parallel is high.

35

IP was from the beginning designed to be a general communication solution. IP technology is now recognised to be cheap and appropriate for supporting both traditional data applications and delay-sensitive real-time applications. To provide expected service for real-time applications, logically (and physically) separate IP networks are used. Each IP network serves only a subset of sensitive applications (e.g. IP telephony) with quite predictable resource requirements. By limiting the range of applications, the total resource demand is predictable, so that the network can be dimensioned using the same traffic models as are used for vertically optimised networks. The benefit of cheap IP equipment is obtained without requiring support for dynamic service provisioning in the IP technology.

Network operators now aim at cutting the overhead cost of maintaining several parallel networks. One current trend is to simplify the infrastructure by running all kinds of applications, with various network service demands, in the same logical IP network (i.e. the Internet). This means that the application heterogeneity in IP networks is increasing.

Wireless access technologies may incur bottlenecks at the edges of the network. One trend is that wireless access technologies for global-area licensed-bands (e.g. GSM, GPRS, UMTS) are migrating from being purely connection-oriented towards applying "IP all the way". These networks will, even in the next generation, be relatively resource-constrained compared with the wired IP networks. Hand-units in these networks traditionally provide real-time applications for human interaction (e.g. voice), but they are now migrating to providing multiple applications. In addition to application heterogeneity and link heterogeneity, the user community is becoming more heterogeneous in terms of service expectations and willingness to pay for the service (e.g. professional users and home entertainment users). All these trends point towards the Internet becoming a ubiquitous multi-service network. Consequently, there are strong commercial reasons for network operators and equipment providers to offer Quality-of-Service (QoS) differentiation in IP networks.

In the research and standardisation bodies the development of QoS support has progressed from providing signalled solutions for the Internet (somewhat resembling the solutions used in vertical networks) to now recognising that more stateless solutions are favourable. RSVP (Resource Reservation Protocol) is the signalling

protocol standardised to set up per-flow quality of service in routers supporting IntServ (Integrated Services) along the path. IntServ is described in Braden et Al, Integrated Services in the Internet Architecture: an Overview, IETF, RFC1633. Each router along the path performs admission control and then recognises the individual application data streams to provide the service expected. It is argued that this model is too complex and does not scale enough to be used in the backbone of the Internet. Others argue that the model scales well enough to be used close to the edges of the network.

10 The scalability problems of per-flow QoS management in routers have resulted in a new approach being taken in the IETF, known as the differentiated services architecture described in Blake et Al, An architecture for Differentiated Services, IETF, RFC2475. The objective is to provide scalable QoS support by avoiding per-flow state in routers. The basic idea is that IP packet headers include a small label (known as the diffserv field) that identifies the treatment (per-hop behaviour) that packets should be given by the routers. Consequently, core routers are configured with a few forwarding classes and the labels are used to map packets into these classes. The architecture relies on packet markers and policing functions at the boundaries of the network to ensure that the intended services are provided.

20 One advantage of differentiated services is that the model preserves the favourable properties that made the Internet successful; it supports scalable and stateless forwarding over interconnected physical networks of various kinds. The standard model is, however, limited to differentiated forwarding in routers and therefore the challenge lies in providing predictable services to end users.

Qualitative services (relatively better than best-effort services, but depending on where the traffic is sent and on the load incurred by others at the time) can be provided by relying only on DiffServ support in routers and resource management mechanisms for semi-static admission control and service provisioning.

30 To provide quantitative (minimum expectation) service, resources must be dynamically administrated by the resource management mechanisms and involve dynamic admission control to make sure that there are sufficient resources in the network to provide the services committed.

The resource manager

The entity performing dynamic admission control is here called a resource manager as e.g. described in Wolf, L.C., Delgrossi, L., Steinmetz, R., Schaller, S., Wittig, H., "Issues of Reserving Resources in Advance", IBM European Network Center Heidelberg, TR 43.9503, 1995. This entity keeps track of the available resources and performs admission control on incoming requests for resources from clients. To perform the admission control the resource manager also stores a history of previously admitted resource reservations. The resource manager takes the decision to admit a new request for resources based on the total amount of available resources, the amount currently reserved by previously reservations and the amount of resources requested. The resources may or may not be scheduled over time. One request may involve admission control on multiple resource repositories that may consist of different types of resources. Bandwidth is the most common type of managed resource.

There are specific requirements for resource management mechanisms. To provide service to end users, they must be aware of network resources and may schedule them for the committed service at any granularity (e.g. for a port range, for aggregate traffic between a pair of subnets, etc). The mechanisms should provide accurate resource control both in leaf domains and in core domains. In leaf domains there may be requirements for very fine granular resource control (e.g. per application data stream), especially at licensed band wireless access where bandwidth is so expensive that spectrum efficiency is the overall goal. The performance must also be sufficient to handle mobility and frequent hand-over.

In core domains, dynamic aggregated resource management (e.g. per destination domain, per port range for IP telephony, etc.) may be provided for scalability reasons. Internet Service Providers (ISPs) need support for negotiating bulk bandwidth with each other by using reservations in advance and time-dependent contracts (e.g. time of day, day of week, etc.) as described in C. Chuah, L. Subramanian, R. Katz, and A. Joseph. "QoS provisioning using a clearing house architecture", In Proceedings of IWQoS 2000, Pittsburgh, PA, June 2000. In enterprise networks, there are often well-provisioned LANs and bottleneck leased lines to interconnect sites. They need support for deploying new internal services (e.g. multimedia conferencing) that require certain amounts of resources, and for

controlling the resource requirements for these applications must be controlled to avoid excessive negative impact on other traffic.

Handling immediate open-ended requests

Many applications, such as IP-telephony, are based on immediate (instantaneous) demands for resources. These applications are also often open-ended in their nature, i.e. the duration of the session is unknown when the resources are requested (e.g. IP-telephony). Reservations that are desired immediately are from now on referred to as immediate reservations and reservations where the duration is not known on beforehand is referred to as open-ended reservations. Thus reservations that are desired "from now on" are referred to as immediate open-ended reservations.

To perform admission control in a system with only this kind of applications, as in **Figure 1** the resource manager needs only to keep track of the current sum of reserved resources (2) to investigate if a new request (1) is allowed to be admitted within the available resources. Notice that in this system there is no need to schedule resources over time (only a counter is needed) and the clients will request resources at the time they are needed. All graphs in the accompanying drawings have resources on the vertical axis and time on the horizontal axis.

Scheduling resources over time

By being able to make reservations in advance, clients can more efficiently plan their network activities and be assured in advance that they will have access to the claimed resources. Reservations made in advance are often time-limited so they can be scheduled over time as described in D. Ferrari, A. Gupta, and G. Ventre. Distributed advance reservation of real-time connections. In Proceedings of NOSSDAV, Lecture Notes in Computer Science, pages 15-26, Durham, New Hampshire, April 1995. Springer.

Thus, the start-time and the stop-time (or alternatively the duration) is often known and these reservations are referred to as in-advance reservations. Notice that the start-time may be very soon in the future (even immediately). In-advance reservations are especially essential for multi-party meetings (e.g. meetings, conferences, lectures, etc.) or for other planned events such as sport-events etc. to make sure that the resources will be available at the time they are required as e.g.

described in S. Bersson, R. Lindell and R. Braden, An architecture for advance reservations in the internet. Technical Report, USC Information Sciences Institute, July 1998. Requesting large amounts of resources over long distances with immediate admission would probably fail when resources are scarce, which would
5 be very disturbing.

Network operators can manually or automatically predict aggregated resource requirements over time and more accurately negotiate long-term aggregate resources between different domains. Even if end users mostly make immediate
10 requests, operators typically plan ahead by negotiating trunk bandwidth with their neighbours. This typically involves scheduling resources over time so new contracts can start at the point when current contracts expire. Contracts may also concern aggregate bandwidths that vary over time (e.g. over the hours of the day or days of the week). Having support for scheduling resources over time is thus crucial for the
15 ability to efficiently providing predictable services over multiple domains and time zones.

The process of scheduling reservations over time is quite straightforward. As shown in **Figure 2**, it is a two-dimensional problem involving resource and time. The
20 admission control involves fitting boxes (2) in a two-dimensional time-resource diagram. The resource manager must keep track of the previously reserved resources (1) over time and see if the new request for resources (2) can be admitted. In this case the request is constrained by a start-time (3) and a stop-time (4) and the height of the box represents the amount of resources that is requested. If there
25 are enough available resources all the time from the start-time to the stop-time (in all affected resource repositories) the request is admitted.

In a system with only constrained reservations with a predefined duration all admitted reservations are set aside and the client will be guaranteed access to the
30 resources as long as the available resources does not change after the reservations was admitted.

State of the art

The most well known architectures for providing QoS over the Internet do not
35 include reservations in advance. The integrated services (IntServ) architecture uses the RSVP protocol to set up reservations on demand when the resources are

required. Most architectures that support in-advance reservations do this by partitioning the resources so that one part of the resources is dedicated for immediate open-ended use (**Figure 1**), and another part is used for time-limited in-advance reservations (**Figure 2**). By partitioning the resources the available
5 resources may not be shared between open-ended and in-advance reservations and the utilisation will be low. As shown in **Figure 3** the resources reserved for the open-ended reservations in the lower part of the figure may not be used for in-advance reservations. The utilisation may be increased slightly by allowing dynamic adjustments of the partitioning level. Notice that this method can be viewed as
10 booking the open-ended reservations using an infinite duration even though the session might not carry on more than a couple of minutes.

One method of mixing immediate open-ended reservations with time-limited in-advance reservations is developed and described in Olov Schelén, Quality of Service
15 Agents in the Internet, Doctoral Thesis, Department of Computer Science and Electrical Engineering, Division of Computer Communication, Luleå University of Technology, Luleå, 1998. In this method, the admission control for immediate open-ended reservations and in-advance reservations was separated but the resources are however shared between the two through signalling information about current
20 bookings. This method is based on statistical predictions of the durations of the immediate open-ended reservations and weighing the risk of having to pre-empt a reservation against the utilisation. This method was optimised to preserve the performance and simplicity of the admission control for immediate open-ended reservations (as in **Figure 1**) by minimising the information that needs to be shared
25 between the two algorithms.

The method described above is illustrated in **Figure 4**. It is based on monitoring the level of admitted in-advance reservations (4) within a look-ahead time (3) when admitting a new open-ended reservation (1). Hence, it is possible to monitor the
30 future resource utilisation by studying the admitted in advance reservations within the look-ahead time. A new immediate reservation (1) is admitted if there is enough resources available for the total sum of immediate open-ended reservations (1+2) looking a predefined time ahead in the future (3), i.e. the look-ahead time. When making admission control for in-advance reservations the immediate (1 and 2)
35 reservations need not be considered because new in-advance reservations are not

admitted within the look-ahead time (3) to protect the previously admitted immediate reservations (1 and 2). The time from making an in-advance reservation until it is to be activated is called the book-ahead time (5), in [1], which is at least the look-ahead time (3).

5

The main problem with mixing open-ended reservations with in-advance reservations is that the duration of the open-ended reservations is not known on beforehand as described in Greenberg, A. G., R. Srikant and W. Whitt, "Resource Sharing for Book-Ahead and Instantaneous-Request Calls," AT&T Labs, 1997. If the resources are not partitioned infinitely as in **Figure 3**, there is always a risk of over-booking if the resources are shared, as described in Schill, F. Breiter, and S. Kuhn, "Design and evaluation of an advance reservation protocol on top of RSVP," in Proc. IFIP 4th International Conf. Broadband Communications, Stuttgart, Chapman & Hall, 1998, pp. 23-- 40.

15

For example, if the open-ended reservation (1) in **Figure 5** lasts longer than the bookings at (2), the resources will be over-booked at (2) and to assure the service for all affected reservations one or more reservations must be pre-empted. There is several ways to choose the reservations to pre-empt based on priority, booking-time or some other policies. In this example the natural decision is to pre-empt the open-ended reservation and favour the reservations made in advance.

20

In the solution presented by Olov Schelén, (illustrated in **Figure 4**) the risk of having to pre-empt an immediate reservation could be adjusted by adjusting the look-ahead time. The problem with this method is however that there is only one look-ahead time for all reservations, which means that all open-ended reservations have the same risk of pre-emption and the expected duration of all reservations must be the same. Thus this method does not support different types of applications simultaneously having different requirements for the risk of pre-emption and/or having different statistical distributions of the expected duration. A look-ahead adjusted for IP-telephony might be a couple of minutes while for a video-conferencing application should have a considerably longer look-ahead. Even if the expected duration would be the same in different applications one application might require a lower risk of pre-emption and thus require a longer look-ahead than another application that might accept a higher risk of pre-emption in exchange for

35

lower block-rate (more likely to be admitted but also more likely to be pre-empted later).

In this method that is described by Olov Schelén, the reservations made in advance are guaranteed the granted resources as long as available resources do not change while immediate open-ended reservations risk pre-emption. Therefore to secure the resources one must make the reservation in advance and in this method the reservation must also be made at least the look-ahead time in advance (5) in **Figure 4**. Since this method does not support time-limited immediate reservations all immediate reservations will be subject to the risk of pre-emption. Another problem is that in-advance reservations may not be modified within the look-ahead time to protect the immediate open-ended. This also implies that soft-state reservations, which will time-out if they are not refreshed, are not supported if the refresh time is less than the look-ahead time.

Thus, an object of the present invention is to provide a method and apparatus for utilising the resources more efficient when reserving resources for immediate and future use.

Summary of the invention

The above-mentioned object is achieved by the present invention according to the independent claims.

Preferred embodiments are set forth in the dependent claims.

By mixing open-ended reservations with in-advance reservations, by utilising a common pool of resources, the resource utilisation will be high. The method according to the present invention allows support for open-ended reservations as well as in-advance and immediate time-limited reservations wherein the resources are guaranteed. It also allows modification of active and "soon to be active" reservations, which in turn permit soft-state reservations. The invention supports different types of applications with varying time-distributions and varying risk of pre-emption. A new concept of open-ended in-advance reservations is also

introduced in the present invention. The present invention supports the possibility to adapt the risk of pre-emption.

The resource manager comprises means for performing the actions as described in the "Resource manager". According to the present invention the resource manager also comprises:

means for reserving resources, means for guaranteeing said resources between said start-time and said stop-time, and means for keeping said resources for the application after said stop-time has expired if said application still needs the resources in order to provide a resource reservation for an open-ended application.

The resource manager may also comprise means for keeping a list of active reservations that have expired said stop-time in order to control the pre-emption of the open-ended reservations if there is not enough available resources.

Furthermore, the resource manager may also comprise means for allowing each application client to set individual start- and stop-time for said application in order to choose said start- and stop-time adapted to said application.

Furthermore, the resource manager may also comprise means for setting individual start- and stop-time for each application in order to choose said start- and stop-time adapted to said application.

Moreover, the resource manager may comprise means for setting said start-time to the current time in order to reserve and guarantee resources immediately.

Moreover, the resource manager may comprise means for setting said stop-time to the current time in order to allocate resources immediately.

Moreover, the resource manager may comprise means for setting said stop-time to infinity in order to guarantee resources during a long time.

In addition, the resource manager may comprise means for basing the charging of said resources on the amount of guaranteed resources.

Brief description of the appended drawings

Figure 1 illustrates handling of immediate open-ended resource requests.

Figure 2 illustrates resource scheduling over time.

5 **Figure 3** illustrates partitioned resources for open-ended and in-advance reservations.

Figure 4 shows admission control based on a look-ahead time.

Figure 5 shows how scheduled in-advance reservations is mixed with open-ended reservations.

10 **Figure 6** shows an immediate open-ended reservation with a start-time and a stop-time according to the present invention.

Figure 7 shows open-ended in-advance reservation according to the present invention.

Figure 8 shows flowchart of the method according to the present invention.

15

Detailed description of preferred embodiments

A first preferred embodiment according to the present invention is based on providing the possibility to book open-ended reservations in the same way and from the same repository as the in-advance and immediate time-limited reservations.

20 This is accomplished by introducing a start-time (1) and a stop-time (2) for the open-ended reservations that may or may not be given by the client as shown in **Figure 6**. The open-ended reservation is booked together with the in-advance reservations (4) between the start-time (1) and the stop-time (2) and is thereby also guaranteed the resources between said start-time (1) and said stop-time (2). The
25 difference between a reservation, as in **Figure 2**, and an open-ended reservation according to the present invention is that after the stop-time (2) has expired, the resources that were reserved between the start- and stop-time may still be used. Thus, this allows open-ended applications to reserve resources. The "open-ended behaviour" of the resource request is indicated in **Figure 6** with the arrow (3). After
30 the stop-time (2) has expired, the resources that were reserved are now subject to the risk of being pre-empted. In this solution only open-ended reservations that last longer than the stop-time (2) may be pre-empted.

A second preferred embodiment according to the present invention allows open-ended in-advance reservations i.e. an open-ended reservation wherein its start-time is set to a time in the future, as shown in **Figure 7**.

5 The open-ended in-advance reservation (1) in **Figure 7** is considered as any other in-advance reservation when making the admission control and booking up the resources but after the stop-time (2) has expired the resources are not released, instead the resources may still be utilised as long as the client desires the resources and as long as there are available resources. This kind of reservations is for
10 example useful when booking resources for a video conference that lasts longer than expected or when booking resources for sport events that often last longer than expected. Even video-on-demand applications might last longer than expected if the user is allowed to pause or rewind the movie. By leaving these kinds of bookings open-ended, the resources will not automatically be released when the
15 stop-time has passed.

The resource manager is required to keep a list of all active open-ended reservations where the stop-time has been passed and to verify that there are available resources to keep these reservations as time goes by. If the current available
20 resources become over-booked, pre-emption of one or more clients using resources where stop-time is expired must be performed. In most cases it is natural to pre-empt one of the clients in this list but there may also exist exceptions such as for emergency calls. I.e., if there are cases with high priority (e.g. emergency) calls in the list, it may be suitable to pre-empt another resource instead. the choice is up to
25 the local policy. In the present invention, there is no special treatment of immediate reservations. There is only the concept of time-limited or open-ended reservations and a mix between the two. Making an immediate reservation (from now) using zero duration (start-time = stop-time = now) gives no guaranties (similar to **Figure 5**) and making an immediate reservation with a duration equal to the look-ahead time
30 in **Figure 4** will provide the same result as the described method by Olov Schelén with the respect to be able to guarantee resources within the look-ahead time.

The solution enables the possibility to have an individual start-time and an individual stop-time for each open-ended reservation provided either by the client or
35 as pre-configured values in the resource manager for different applications. Charging may, or may not be based on the amount of guaranteed resources (the

duration between the start and stop-times). If the client wants a lower risk of pre-emption he is required to request a resource reservation for a longer duration. On the other hand, a request for a longer duration may not be accepted. Thus, the duration may be chosen with regard to the desired blocking probability, a
5 reservation with shorter duration will more likely be admitted and a reservation with zero duration will always be admitted since no resources are set aside but the risk of pre-emption will then be very high. The present invention supports immediate time-limited reservations, which allows soft-state reservations, and allows guaranteed resources for immediate reservations. A soft-state reservation, is
10 a time-limited reservation that is released after a while, wherein the release is controlled by e.g. a timer. If a client wants to keep a soft-state reservation, continuously repeated refresh operations are required. The difference between a soft-state reservation and an open-ended reservation, is that an open-ended is keeping its resources until they are released by an active action.

15 Figure 8 shows a flowchart of a method for reserving resources for an application in an IP-network, according to the invention in a general mode. The resources are reserved and controlled by a resource manager, wherein the resource manager is located within a server or a router within said IP-network.

20 The method comprises the following steps:

801. A client requests a reservation of resources for an application from a resource manager and sets a start-time and stop-time.

802. The resource manager either admits or rejects the request.

25 **803.** If the request is admitted, the resource manager reserves and guarantees resources from the start-time onwards, to the stop-time.

804. When the stop-time expires, the reserved resources are kept if said application still needs said reserved resources and if it is enough available resources.

30 The resource reservation method according to the present invention is implemented by a computer program product, wherein the computer program product may be implemented in a resource manager, wherein the resource manager is located within a router or a server within an IP network.

35 The present invention is not limited to the above-described preferred embodiments. Various alternatives, modifications and equivalents may be used. Therefore, the

above embodiments should not be taken as limiting the scope of the invention, which is defined by the appending claims.